

# Multiple Categorization of Search Results

Catherine Blake, M.S. and Wanda Pratt, Ph.D.

Information and Computer Science Department

University of California, Irvine

{cblake, pratt}@ics.uci.edu

*The number of publications available to physicians and patients is increasing at an alarming rate. Although users can use tools to assist in reformulating their query, this approach is ineffective when their information needs are imprecise or many documents are relevant. The ranked list presentation of documents provides little or no information relating documents to the initial query or to each other. We argue that information overload can be reduced if documents are placed in categories that (a) relate to the initial query and (b) contain a manageable number of documents. Dynamic Categorization is a knowledge-based approach that satisfies (a), however the number of documents in a category may still be large. We demonstrate that using the same approach to re-categorize the documents in large categories reduces the number of documents in the sub-category, and maintains a clear relationship to the initial query.*

## Introduction

MEDLINE currently contains more than 11 million bibliographic references. These refereed publications from over 3800 different journals grow at the rate of approximately 31,000 new entries each month.<sup>1</sup> As the number of medical publications continues to increase, both users with little medical expertise and experts need search tools to help them remove irrelevant documents and to navigate through the relevant documents.

Existing information retrieval systems assume that: (a) an information need can be satisfied with a small number of documents; (b) relevant documents are independent; and (c) an information need can be clearly articulated. We argue that these assumptions rarely apply, particularly in the medical domain. Consider the number of publications produced during the development of a new drug treatment program. The drug must go through varying levels of controlled trials before gaining Federal Drug Administration (FDA) approval. Due to the imprecise nature of medicine, multiple studies, on different populations are required to demonstrate the performance of the new drug treatment and to ensure that the medical community understands and discloses potential side effects. This characteristic is contrary to the assumption that a set of relevant documents is small. The number of documents on a particular topic gives a user implicit information regarding the maturity of a drug treatment.

The medical domain is notorious for its complex vocabulary. It is unreasonable to expect patients, with no formal education in medicine, to be able to articulate their information requirements clearly.

Traditional information retrieval systems are ineffective in the medical domain because they display relevant documents to a user as a ranked list, thus the searcher has to read the entire list of relevant documents. This situation may be feasible with a small number of documents, but medical queries result in a large number of relevant documents. A ranked list does not provide the user with information on how the documents are related. Therefore, users must read (and understand) the entire list of relevant documents before they become familiar with the different treatment options available.

A traditional approach to information overload is to specify additional constraints in the hope of reducing the number of documents. Patients and physicians can specify constraints on either search terms or meta-data such as the author, date and type of publication. Patients unfamiliar with medical terminology are unlikely to specify their information need with enough detail to make the number of documents returned manageable, without also eliminating many relevant documents. Even physicians who do understand the terminology have difficulty translating their clinical information need into a query<sup>2</sup>. Meta-data is generally orthogonal to the information request. An alternative solution to the information overload problem is to cluster the documents and present the user with these clusters. The challenge with this approach is that providing the user with a brief meaningful description of the topics discussed in the cluster is non-trivial. Although statistically optimal, clusters may not be optimal with respect to the query.

## Dynamic Categorization

**Dynamic Categorization** is a knowledge-based approach that arranges documents into a hierarchy of categories related to a users initial query.<sup>3,4</sup> It provides the user with the number of documents in each category to assist them in navigating through the hierarchy. Users satisfies their information need by selecting a document set, based on content, rather than based on meta-data such as the publication date or type.

**DynaCat** is an implementation of Dynamic Categorization for the medical domain. It automatically creates a hierarchy of categories pertinent to the query and assigns the appropriate documents to each category.

As opposed to relevance-ranking tools, the purpose of DynaCat is not to separate irrelevant from relevant documents, but rather to organize relevant documents and relates the documents back to the user's initial query. This approach can provide such capabilities because it is based on a representation of the documents that is semantically richer than that used by most search systems.

The semantics in dynamic categorization stem from two types of models: (1) a small **query model** that contains knowledge about what types of queries users make, and how search results from those queries should be categorized, and (2) a large domain-specific **terminology model** that connects individual terms to their corresponding general concept or semantic type (e.g. *aspirin's* semantic type is *pharmacologic substance*). DynaCat uses the Unified Medical Language System (UMLS)<sup>5</sup>, which provides semantic information on over 500,000 biomedical terms.

**Table 1. Medical query types and their typical forms**

Query Type	Form of Question
preventive-actions	What can be done to prevent <problem>?
risk-factors	What are the risk factors for <problem>?
tests	What are the diagnostic tests for <problem>?
symptoms	What are the warning signs and symptoms for <problem>?
diagnoses	What are the possible diagnoses for <symptoms>?
treatments	What are the treatments for <problem>?
problems	What are the adverse effects of <treatment>?
prognostic-indicators	What are the factors that influence the prognosis for <problem>?
prognoses	What is the prognosis for <problem>?

DynaCat's query model maps between the types of queries a user may enter and the criteria for generating categories that correspond to the user's query. **Query types** are high-level representations of the user queries that are independent of disease-specific terms; therefore, many queries have the same query type. DynaCat contains nine query-types: tests, risk factors, treatments, prognoses, prognostic-indicators, preventive-actions, diagnosis, symptoms and side effects. Each query type is mapped to the criteria that

specify the conditions that must be satisfied for a document to belong to that type of category. DynaCat takes advantage of the keywords or Medical Subject Headings (MeSH) terms assigned to medical journal articles. DynaCat must prune the irrelevant keywords from the list of potential categories because many of a document's keywords do not correspond to the user's query.

DynaCat is different from web search engines in many ways. Firstly, the information being categorized is from refereed journals, not web pages. The key function of DynaCat is to organize relevant documents, not to distinguish between relevant and irrelevant documents and lastly the categories are dynamically generated, not from manually defined categories.

### Detailed Scenario

Consider a user who wants information on how to prevent breast cancer, and expresses her information need as "Breast Cancer prevention". PubMed returns 8474 documents for such a query. To gain a high level understanding of the preventive methods discussed in the search results, the user must sift through the entire ranked list of documents. Clearly, this approach is unrealistic. Presenting documents as a hierarchy enables a user to gain a high level understanding of the approaches taken in response to their query, without having to read all the documents.

In the DynaCat framework, this user's information need corresponds to a preventive-action query-type. This user would enter breast cancer in the appropriate query type on the DynaCat search interface and background knowledge regarding the query type would transform the query into "breast cancer/Health Service, Preventive". Using the 5255 documents returned from PubMed using the transformed query, DynaCat generates the top-level categories shown in figure 1. The categories are based on the documents retrieved and the query model, thus the top-level view would be different if the user were looking for preventative actions of another medical condition.

The number of documents in each category provides the user with implicit information about the maturity of that area of study. If a user was not prepared to use surgical procedures to prevent breast cancer, they can disregard 170 documents immediately. Adding this selection criterion to traditional information retrieval systems would require that a user explicitly state the exact name of each surgical procedure to exclude. Such an assumption is inappropriate for users who are still learning about their options. Pruning too early may exclude documents that compare surgical procedures to a topic that was of interest to the user (e.g. drug therapies).

---

Behavior and Behavior Mechanisms (977 refs)  
 Chemicals and Drugs (17 refs)  
 Diagnostic Techniques and Procedures (3285 refs)  
 Nutrition (22 refs)  
 Preventive Health Services (876 refs)  
 Surgical Procedures, Operative (170 refs)  
 Not Otherwise Specified (1112 refs)

---

**Figure 1 — Top-Level Categories for Breast Cancer Prevention Scenario**

---

Another advantage of DynaCat is that categories with a relatively small number of documents are included in the hierarchy. In a ranked list presentation, the 17 documents related to *Chemicals and Drugs* might be lost among the 5238 other documents related to other preventive approaches.

User studies have shown that DynaCat does help people find answers to common types of medical queries more efficiently and easily than they could with the standard relevance-ranking systems.<sup>6</sup> This study showed that users found categories with many documents to be less useful.

### Multiple Categorization

From a set of relevant documents and a query type, DynaCat produces a hierarchy of categories related to the initial query. The problem we address in this paper is what to do when the number of documents in a category becomes unmanageable, where we have defined unmanageable to be over 50 documents. We envision that this would be user-defined.

Multiple Categorization partitions the relevant documents in an unmanageable category into sub-categories by applying one of the query types listed in table 1. The type of documents in the unmanageable category determines the choice of query type. This approach not only reduces the number of documents in unmanageable categories, but also maintains a clear relationship between the documents and the initial query.

We start with the categorization produced by DynaCat. Now consider a category *C* in this hierarchy that contains an unmanageable number of documents. We have written a function that uses the MeSH terms associated with the documents in *C* (via their semantic types) to indicate the type of documents contained in the category. Once we know the type of documents in *C*, we are able to use background knowledge to propose a meaningful re-categorization strategy. For example, if all of the documents in the unmanageable category *C* discuss treatments, then we would re-categorize the documents in this category *C* based on the query: *What are the risk factors associated with the treatments in C?* If all of the documents in *C* dis-

cussed *prognostic-indicators*, then we would propose a re-categorization based on the query: *What is the prognosis of C?*

Once we have a set of candidate query types for further categorization, we can re-run DynaCat. This results in a new classification hierarchy based on the subset of relevant documents in the category *C* and the new query type. We need remove the MeSH term *C* from the set to avoid generating the same category and we do not suggest a query that has previously been used. This process is outlined in Figure 2.

- 
- (1) Apply DynaCat using the initial query and set of relevant documents
  - (2) For each category *C* with an unmanageable number of documents
    - (2a) Determine the type of documents in *C*.
    - (2b) Suggest a query type that is appropriate for the type of documents found in 2a)
    - (2c) Apply DynaCat using the documents in the *C* and the new query type suggested in 2b)
- 

**Figure 2 — The Multiple Categorization strategy.**

---

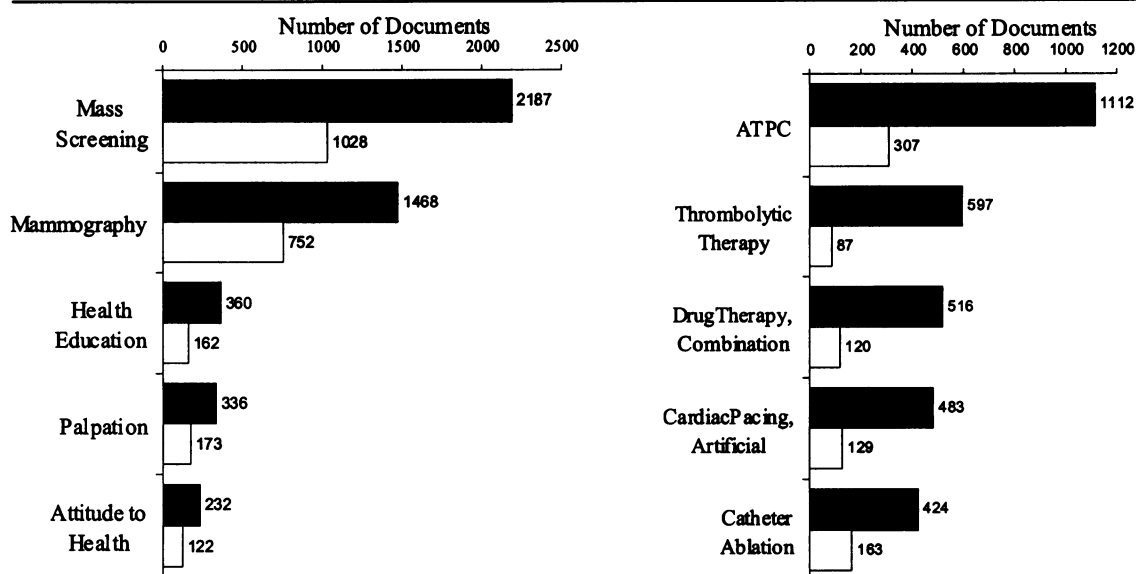
An issue associated with re-categorization is that generating many categories, each with a single document is just as unmanageable as a single category with all the documents. To avoid excessive partitioning we have modified DynaCat to provide sub-categories only when the number of documents in the current category exceeds five. We set this limit based on psychological studies on memory.<sup>7</sup>

An alternative approach to re-categorization is to use meta-data (such as publication type or substance). Meta-data may not be available for all documents (this may result in a large number of documents being placed in the 'Not Otherwise Specified' category) and does not take into account the initial query or the type of documents in the category.

### Evaluation

We evaluated our re-categorizing strategy based on how well it reduced the number of documents in the largest categories (Figure 2), and how it changed the distribution of the number of documents in each category (Figure 3). For these experiments, we define *manageable* as those categories that have fewer than fifty documents. We evaluated Multiple Categorization on two queries: *How do I prevent Breast-Cancer?* and *What are the symptoms of heart disease?*

Figure 2 shows the effect of re-categorization on the largest categories. Figure 2A displays the 5 categories with the largest number of documents for our first query: *How do I prevent Breast-Cancer?* The



**Figure 2 - Multiple Categorization reduced the number of documents in the unmanageable categories.** The largest 5 categories in original categorization (Black) are reduced after re-categorization (white). The average reduction for all unmanageable categories (except for *Not Otherwise Specified*) was 52% in the Breast Cancer scenario and 73% in the Heart Disease scenario. The scenarios has 21 and 17 unmanageable categories respectively.

largest category, *Mass Screening* contained 2187 documents. This category name (a MeSH term) maps to both the test and prognostic-indicators query-type. We used background knowledge to infer that re-categorizing on risk factors is appropriate. The number of documents in the largest category using this new query was 1028, a 53% reduction. All categories (except *Patient Education* that did not return a valid query type) had a query type of *test* or *treatment*. Thus, we re-categorized all of the unmanageable categories based on the *risk factor* query type. The average reduction for the largest sub-category was 52%.

Now consider a user who needs information on the warning signs and symptoms of Heart Disease. Expressing this information need as "heart disease symptoms" in PubMed results in more than 260,000 documents. Using background knowledge of the *symptoms* query type, DynaCat transforms the query into "heart disease/Pathological Conditions, Signs and Symptoms". This results in 9734 documents.

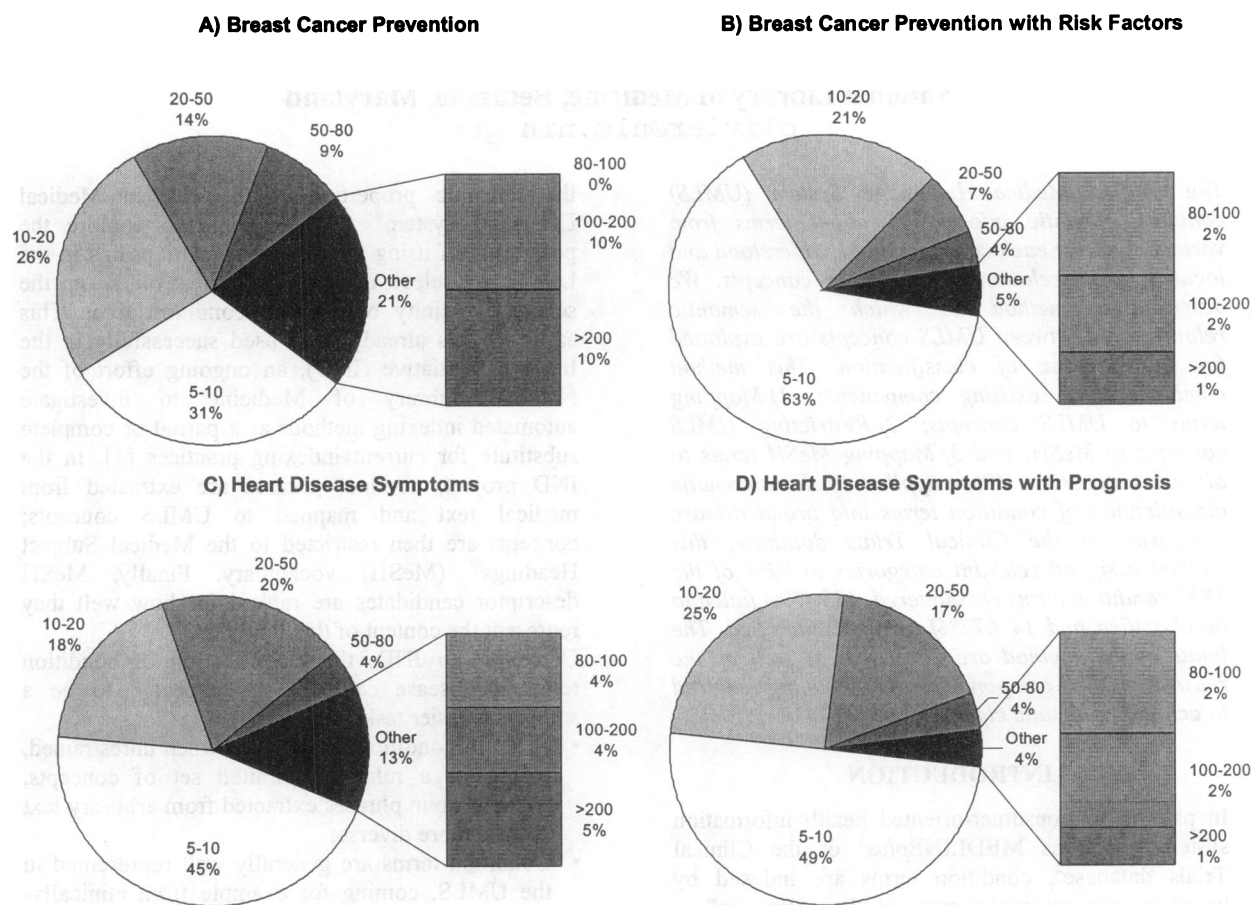
All unmanageable categories in the Heart Disease query were of the *prognostic-indicators* query type. In addition, seven categories were also *preventive-action* query types. We can infer that a re-categorization based on the *prognosis* query type would be appropriate. The average reduction in unmanageable categories for this query was 71%.

In addition to reducing the number of documents

in the largest categories, we should increase the percentage of categories with between 5 and 20 documents and reduce the number of categories with over 50 documents. Figure 3 displays the degree to which our approach was successful in this goal. The number of categories containing less than 20 documents was increased from 57% to 84% and from 63% to 74% in the Breast Cancer and Heart Disease queries respectively. We reduced the number of unmanageable categories from 30% to 9% in the breast cancer query and from 13% to 4% in the heart disease query.

## Conclusions

Existing information retrieval systems that produce a ranked-list of relevant documents do not address characteristics that exist in the medical domain. There will be many documents relevant to a query, which are dependent. For a novice it is difficult to articulate a precise information need. DynaCat provides physicians and patients with a way to navigate through many relevant documents. This approach however may still result in categories with large numbers of documents. We have demonstrated a technique that focuses on categories with many documents and further partitions these documents in a meaningful way. We argue that re-categorizing based on the type of documents in the large category relates better to the initial query.



**Figure 3 - Distribution of the number of documents in a category. A, B. Breast cancer prevention query before and after re-categorization. C, D Symptoms of Heart Disease query before and after re-categorization. An improvement in performance would result in a greater number of categories with between 5 to 20 documents, that is the lightest most segments.**

Although these results are encouraging, we need to conduct further evaluations to determine the degree to which re-categorization makes it easier for physicians and patients to satisfy their information need. We are currently developing a user interface for this purpose. It will incorporate user feedback, and allow a user to select among the recommended re-categorization strategies. Our approach can be applied recursively to the re-categorization. We plan to include this recursive re-categorization in our user-based evaluations.

#### Acknowledgements

We would like to thank Craig Evans for his comments on earlier drafts of this paper.

#### References

1. NLM. NLM Online Databases and Databanks. [Online] [http://www.nlm.nih.gov/pubs/factsheets/online\\_databases.html#medline](http://www.nlm.nih.gov/pubs/factsheets/online_databases.html#medline). 1999.

2. Osheroff, JA, Bankowitz, RA. Physicians' Use of Computer Software in Answering Clinical Questions. *Bull Med Libr Assoc* 1993;81(1):11-9
3. Pratt W. Dynamic Categorization: A Method for Decreasing Information Overload. PhD Thesis. Medical Information Sciences, Stanford University.
4. Pratt W, Hearst MA, Fagan LM. A Knowledge-Based Approach to Organizing Retrieved Documents. In: *AAAI '99: Proc. Sixteenth National Conference on Artificial Intelligence*; Orlando, FL; 1999.
5. NLM. The UMLS Metathesaurus Fact Sheet. [Online] <http://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html>. 1999.
6. Pratt W, Fagan L. The Usefulness of Dynamically Categorizing Search Results. *Journal of the American Medical Informatics Association (JAMIA)* 2000;(In Press).
7. Miller GA. The magical number seven, plus or minus two. *Psychological Review* 1956;63:81-97.